|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ **Информатика и системы управления**

КАФЕДРА **Компьютерные системы и сети (ИУ6)**

НАПРАВЛЕНИЕ ПОДГОТОВКИ 09.03.01 **Прикладная информатика**

**Отчет**

**по лабораторной работе № 5**

**Дисциплина:** Разработка приложений на языке C#

**Название работы:** Файлы



Студент гр. ИУ6-72Б **\_\_**01.10.2022**\_\_\_\_\_\_\_ \_\_\_**И.С. Марчук\_\_

(Подпись, дата) (И.О. Фамилия)

Преподаватель  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** А.М. Минитаева

(Подпись, дата) (И.О. Фамилия)

Москва, 2022

**Цель работы:** изучить основы работы с текстовыми файлами на языке программирования C#.

**Задание:**

1. Программа должна быть разработана в виде приложения Windows Forms на языке C#. По желанию вместо Windows Forms возможно использование WPF (Windows Presentation Foundation).

2. Добавить кнопку, реализующую функцию чтения текстового файла в список слов List<string>.

3. Для выбора имени файла используется класс OpenFileDialog, который открывает диалоговое окно с выбором файла. Ограничить выбор только файлами с расширением «.txt».

4. Для чтения из файла рекомендуется использовать статический метод ReadAllText() класса File (пространство имен System.IO). Содержимое файла считывается методом ReadAllText() в виде одной строки, далее делится на слова с использованием метода Split() класса string. Слова сохраняются в List<string>.

5. При сохранении слов в список дубликаты слов не записываются. Для проверки наличия слова в списке используется метод Contains().

6. Вычислить время загрузки и сохранения в список с использованием класса Stopwatch (пространство имен System.Diagnostics). Вычисленное время вывести на форму в поле ввода (TextBox) или надпись (Label).

7. Добавить на форму поле ввода для поиска слова и кнопку поиска. При нажатии на кнопку поиска осуществлять поиск введенного слова в списке Слово считается найденным, если оно входит в элемент списка как подстрока (метод Contains() класса string).

8. Добавить на форму список (ListBox). Найденные слова выводить в список с использованием метода «название\_списка.Items.Add()». Вызовы метода «название\_списка.Items.Add()» должны находится между вызовами методов «название\_списка.BeginUpdate()» и «название\_списка. EndUpdate()».

9. Вычислить время поиска с использованием класса Stopwatch. Вычисленное время вывести на форму в TextBox или Label.

Листинг программы:

Program.cs:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace \_4

{

static class Program

{

/// <summary>

/// Главная точка входа для приложения.

/// </summary>

[STAThread]

static void Main()

{

Application.EnableVisualStyles();

Application.SetCompatibleTextRenderingDefault(false);

Application.Run(new Form1());

}

}

}

Form1.cs:

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.IO;

using System.Diagnostics;

namespace \_4

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

Width = 300;//задаем нач размер

Height = 300;

}

readonly List<string> list = new List<string>();//создаем лист

private void button1\_Click(object sender, EventArgs e)//кнопка открытия файла

{

#region настройка OpenFileDialog

OpenFileDialog opendial = new OpenFileDialog();//экз. класса

opendial.InitialDirectory = "";//указываем начальную дерикторию

opendial.Filter = "Файлы txt \*.txt|\*.txt";//ставим фильтир только на txt

opendial.ReadOnlyChecked = true;// режим только для чтения

#endregion

Stopwatch stopWatch = new Stopwatch();//экз. класса

if (opendial.ShowDialog() == DialogResult.OK)//если диал. окно открыто

{

stopWatch.Start();//запускаем отсчет времени

string str = File.ReadAllText(opendial.FileName);//записываем в переменную стринг текст из файла

string[] s\_mas = str.Split();//записываем в массив стринг слова из str

foreach(string o in s\_mas)

if (!list.Contains(o) && o != "")//если в листе нет такой строки и она не путая

list.Add(o.ToLower());//заносим в лист делая все слова маленькими

stopWatch.Stop();//останавливаем счетчик времени

textBox1.Text = stopWatch.Elapsed.ToString();//вводим в текстбокс сколько по времени это считалось

#region появление кнопок

textBox2.Visible = textBox3.Visible = true;

listBox1.Visible = true;

label2.Visible = true;

button2.Visible = true;

Width = 850;

Height = 400;

#endregion

}

}

private void button2\_Click(object sender, EventArgs e)//кнопка поиска

{

Stopwatch stopWatch = new Stopwatch();//экз. класса

stopWatch.Start();//запускаем отсчет времени

string s = textBox3.Text.ToLower();//добавляем содер текстбокса маленькими буквами

if (list.Contains(s))//если слово есть в списке

{

foreach (string o in list)//ищем его

if (o == s)

{

listBox1.BeginUpdate();//

listBox1.Items.Add(o);// добавляем слово

listBox1.EndUpdate();//

stopWatch.Stop();//останавливаем подсчет времени

textBox2.Text = stopWatch.Elapsed.ToString();//выводим время

break;

}

}

else

{

listBox1.Items.Add("Слова нет в списке");//если слова нет в списке

stopWatch.Stop();//останавливаем подсчет времени

textBox2.Text = stopWatch.Elapsed.ToString();//выводим время

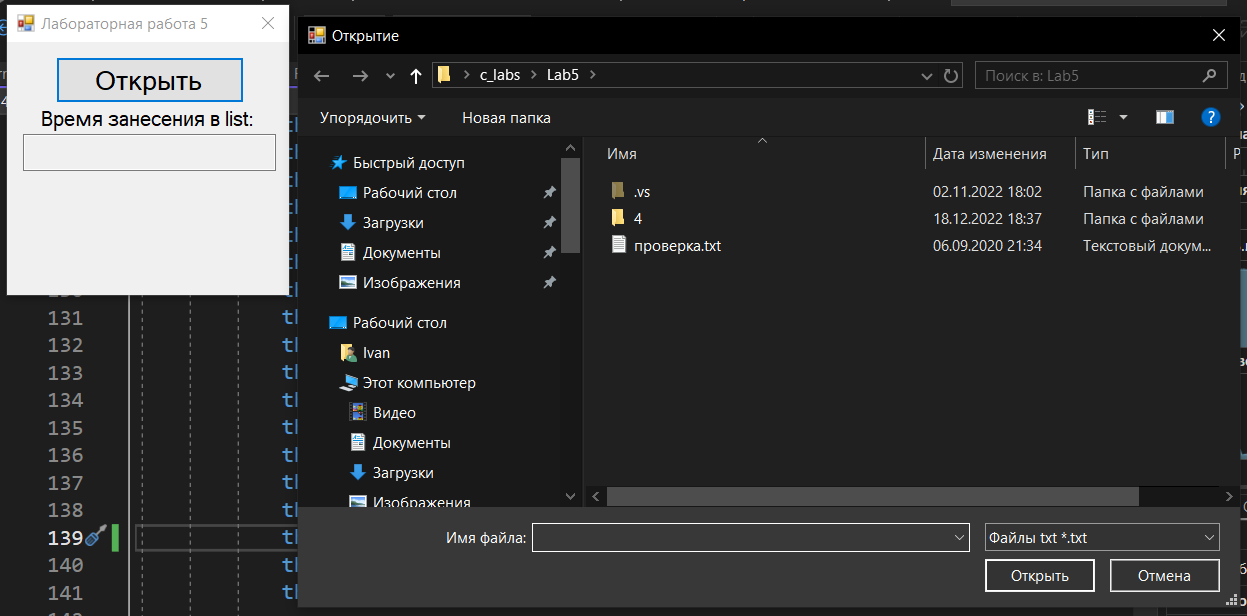
}

}

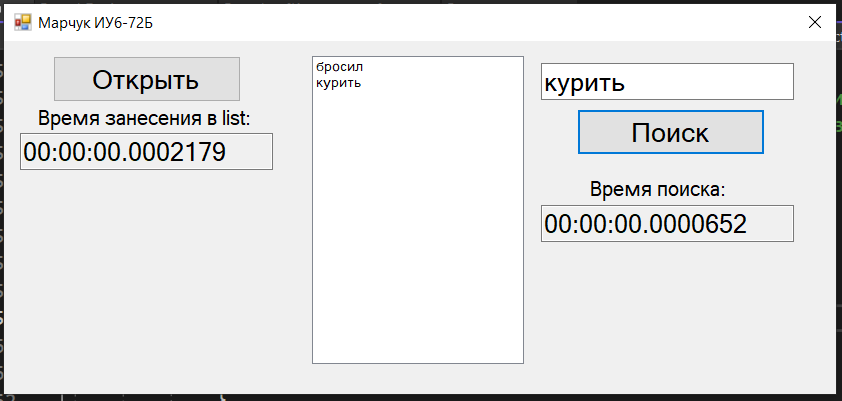
}

}

Работа программы показана на рисунке 1.



*Рисунок 1 – Главное окно и окно выбора файла*



*Рисунок 2 – Работа программы (время поиска для слова «курить» в текстовом файле составило 652 десятимилионных секунды)*

**Вывод:** в процессе выполнения лабораторной работы были изучены средства работы с текстовыми файлами и строками: способ открытия файла определенного расширения, чтение файла в одну строку, дробление строки в слова по разделителям. А также был использован инструмент stopWatch, позволяющий засекать время выполнения кода.